Automatismes – Séance 12

Lycée Les 3 Sources ; NSI T^{le}

Question 1

Donner les instructions permettant de réaliser les actions suivantes :

- Créer une liste Python appelée `valeurs` contenant 5, 8, 2 et 4.
- Rajouter 15 à la fin de la liste.
- Affecter la première valeur de la liste à une variable `premier`.
- Tripler la valeur de l'élément d'indice 2 de la liste.

Question 2

La fonction récursive `est_present` est telle que `est_present(liste, mot, debut, fin)` renvoie l'indice du texte `mot` dans la liste `liste` s'il se trouve entre les indice `debut` (inclus) et `fin` (exclus), et **`None`** s'il n'y est pas.

```
def est_present(liste, mot, debut, fin):
    if debut > fin:
        return False
    milieu = (debut + fin) // 2
    if liste[milieu] > mot:
        return est_present(liste, mot, 0, milieu-1)
    elif liste[milieu] < mot:
        return est_present(liste, mot, milieu+1, fin)
    else:
        return True
```

Question 3

Que se passe-t-il avec `est_present(['b', 'c'], 'c', 0, 2)` et corriger la fonction. Il faut qu'elle soit correcte si on cherche 'a', 'b', 'c' et 'd'.

```
def est_present(liste, mot, debut, fin):
    if debut > fin:
        return False
    milieu = (debut + fin) // 2
    if liste[milieu] > mot:
        return est_present(liste, mot, debut, milieu-1)
    elif liste[milieu] < mot:
        return est_present(liste, mot, milieu+1, fin)
    else:
        return True
```

Correction

Question 1

Donner les instructions permettant de réaliser les actions suivantes :

- Créer une liste Python appelée `valeurs` contenant 5, 8, 2 et 4.
- Rajouter 15 à la fin de la liste.
- Affecter la première valeur de la liste à une variable `premier`.
- Tripler la valeur de l'élément d'indice 2 de la liste.

Question 1

Donner les instructions permettant de réaliser les actions suivantes :

- Créer une liste Python appelée `valeurs` contenant 5, 8, 2 et 4.
- Rajouter 15 à la fin de la liste.
- Affecter la première valeur de la liste à une variable `premier`.
- Tripler la valeur de l'élément d'indice 2 de la liste.

```
>>> valeurs = [5, 8, 2, 4]
>>> valeurs.append(15)
>>> premier = valeurs[0]
>>> valeurs[2] = valeurs[2]*3
```

Question 2

La fonction récursive `est_present` est telle que `est_present(liste, mot, debut, fin)` renvoie l'indice du texte `mot` dans la liste `liste` s'il se trouve entre les indice `debut` (inclus) et `fin` (exclus), et **`None`** s'il n'y est pas.

Question 2

La fonction récursive `est_present` est telle que `est_present(liste, mot, debut, fin)` renvoie l'indice du texte `mot` dans la liste `liste` s'il se trouve entre les indice `debut` (inclus) et `fin` (exclus), et **`None`** s'il n'y est pas.

Dans le premier appel, il faut mettre `debut` à la place de `0`.

Question 3

Que se passe-t-il avec `est_present(['b', 'c'], 'c', 0, 2)` et corriger la fonction. Il faut qu'elle soit correcte si on cherche `'a'`, `'b'`, `'c'` et `'d'`.

Question 3

Que se passe-t-il avec `est_present(['b', 'c'], 'c', 0, 2)` et corriger la fonction. Il faut qu'elle soit correcte si on cherche `'a'`, `'b'`, `'c'` et `'d'`.

On se retrouve avec l'appel `est_present(['b', 'c'], 'c', 2, 2)` qui provoque une erreur.

Il faut modifier la condition de base en `debut >= fin` et le premier appel récursif en `est_present(liste, mot, debut, milieu)` pour être sûr que `fin` ne soit pas égal à l'indice du mot cherché.

